A Brief Introduction to FEniCS Installation[1]

Ran Wang[2]

August 1, 2023

[1] https://github.com/OsbertWang/install-fenics-guide-zh-cn

[2] ranwang.osbert@outlook.com

Abstract

This document records my experience of installing FEniCS.

At the very beginning, I installed FEniCS under Ubuntu 20.04 in WSL. As I don't have much experience with Ubuntu, there might be some mistakes in my record. Besides, the development of FEniCS has been stopped, and its successor, FEniCSx, is not yet stable. This undeniably causes uncertainty for users.

On October 23, 2021, I tried to install Docker. Therefore, I have added the steps to install FEniCS under Docker.

On October 28, 2021, I formally wrote the installation steps of FEniCSx into the manual. Following the official suggestions at the time, I only wrote down the installation steps under Docker.

On March 25, 2023, I installed Ubuntu 22.04 in WSL under Windows 11 and attempted to install FEniCSx.

On April 13, 2023, I attempted to install FEniCSx in Anaconda.

In fact, the process of directly installing FEniCSx in Ubuntu is similar to what is introduced in this manual, users can directly refer to the relevant content.

Currently, the FEniCSx tutorial can be found on the official FEniCSx website. During reading, you can submit Github issues at any time in the upper right corner of the page. Besides the tutorial, users can also read the FEniCSx Project Documentation to learn some usage.

I hope this manual will facilitate users.

Contents

Chapter 1

Installing FEniCSx in WSL

1.1 Install WSL

There are many tutorials on how to install WSL on the Internet, for instance, the official Microsoft documentation. However, due to the fact that the mainland network cannot freely access the WSL json file which is backed up at raw.githubusercontent.com, the direct installation using

```
wsl --install
```

fails. Therefore, alternative plans have to be adopted, such as the older manual installation steps or modifying the hosts file.

Specifically, find the IP corresponding to raw.githubusercontent.com on some IP query websites, for example, the one I found is

```
185.199.111.133 raw.githubusercontent.com
```

Next, open the following with administrative privileges

```
C:\Windows\System32\drivers\etc\hosts
```

and add the queried IP to the end of the file. Now, if you execute

```
wsl -l -o
```

you can see the currently permitted WSL distributions to install

```
NAME FRIENDLY NAME
Ubuntu Ubuntu
Debian Debian GNU/Linux
kali-linux Kali Linux Rolling
Ubuntu-18.04 Ubuntu 18.04 LTS
Ubuntu-20.04 Ubuntu 20.04 LTS
Ubuntu-22.04 Ubuntu 22.04 LTS
OracleLinux_8_5 Oracle Linux 8.5
OracleLinux_7_9 Oracle Linux 7.9
SUSE-Linux-Enterprise-Server-15-SP4 SUSE Linux Enterprise Server 15 SP4
openSUSE-Leap-15.4 openSUSE Leap 15.4
openSUSE-Tumbleweed openSUSE Tumbleweed
```

1.2 Moving the Installation Location of WSL

Considering that some users have limited system disk space, and according to the official Microsoft documentation, WSL can only be installed on the system disk. Therefore, it is possible to consider moving the

installation location of WSL to another drive. This section is sourced from StackExchange. The method involves operations on the main system and the subsystem.

The first step is to create a new location. Suppose the new location is on the `X:\` drive. Run

```
wsl -l
```

in cmd to see the currently installed WSL distributions, for example

```
Windows Subsystem for Linux Distributions:
Ubuntu (Default)
```

Then, execute

```
lsb_release -a
```

in bash to check the actual version of Ubuntu in the WSL, for example

```
No LSB modules are available.
Distributor ID: Ubuntu
Description: Ubuntu 22.04.2 LTS
Release: 22.04
Codename: jammy
```

Next, execute in cmd

```
mkdir X:\WSL\instances\Ubuntu2204
mkdir X:\WSL\images
cd X:\WSL\images
```

to create a new location. Here, we name it `Ubuntu2204` because the actual version of Ubuntu is 22.04, but you can use other names.

The second step is to export the original WSL distribution and import it to the new location. In cmd, execute

```
wsl --export Ubuntu ubuntu.tar
wsl --import Ubuntu2204 X:\WSL\instances\Ubuntu2204 ubuntu.tar --version 2
```

The first line's `Ubuntu` refers to the default system WSL distribution. This command exports the original distribution into a compressed package. The second line's `Ubuntu2204` is the designated future WSL distribution, which means importing the previously exported package into a new WSL distribution.

The third step is to launch the new distribution and set it as default. In cmd, execute

```
wsl ~ -d Ubuntu2204
```

to enter the WSL system of the `Ubuntu2204` distribution. In the current bash, execute

```
sudo -e /etc/wsl.conf
```

and input the following

```
[user]
default=<your_username>
```

`<your_username>` is a user-defined WSL username, personally I used the original username from the `Ubuntu` distribution. Then press Ctrl+X, Y, Enter in sequence to save and exit. Exit `bash`, and in `cmd`, execute

```
wsl --terminate Ubuntu2204
wsl ~ -d Ubuntu2204
```

If everything is normal, you can set `Ubuntu2204` as the default WSL distribution by executing in `cmd`

```
wsl --set-default Ubuntu2204
```

Now, if you execute in bash

```
echo $WSL_DISTRO_NAME
```

and the return result

```
Ubuntu2204
```

indicates that everything is normal.

The fourth step is to delete the old distribution. By executing in cmd

```
wsl --unregister Ubuntu
```

you will delete the old distribution `Ubuntu`. Now, if you execute in cmd

```
wsl -l
```

you will see

```
Windows Subsystem for Linux Distributions:
Ubuntu2204 (Default)
```

In fact, moving the entire WSL installation location has another benefit. According to the official Microsoft documentation, storing project files directly on the WSL drive can improve performance speed.

1.3 Choosing an Editor

Although Windows 11 now allows visualization for WSL, I still prefer using VS Code. Detailed introductions can be found in the official Microsoft document, which primarily utilizes the Remote Development.

1.4 Switching Ubuntu Sources

From this section onwards, unless otherwise stated, all command line operations are to be performed in the WSL's bash.

Given the specific network characteristics in mainland China, it is highly recommended that users change Ubuntu's source before installing FEniCSx. Here, I use the Tsinghua University mirror. Of course, there are other mirrors, such as the University of Science and Technology of China mirror.

Execute

```
sudo cp /etc/apt/sources.list /etc/apt/sources.list.bak
```

to back up the sources.list file. Then execute

```
cp /etc/apt/sources.list ~/sources.list
```

to copy a sources.list to the user folder.

```
code ~/sources.list
```

to open the file, replace the file content with the following content[1]

```
# The source mirror is commented out by default to speed up apt update, you can uncomment it as needed
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe multiverse
```

[1] Specific content may vary depending on the Ubuntu version.

```
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe multiverse

# deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe multiverse
# # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe multiverse

deb http://security.ubuntu.com/ubuntu/ jammy-security main restricted universe multiverse
# deb-src http://security.ubuntu.com/ubuntu/ jammy-security main restricted universe multiverse

# Pre-release software source, not recommended to enable
# deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe multiverse
# # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe multiverse
```

Then execute

```
sudo cp ~/sources.list /etc/apt/sources.list
```

to replace the original sources.list file. After that, execute

```
sudo apt update && sudo apt upgrade
```

to switch sources and update. If there's an error, you can execute

```
sudo cp /etc/apt/sources.list.bak /etc/apt/sources.list
```

to restore the file.

1.5 Adding PPA

Execute

```
sudo add-apt-repository ppa:fenics-packages/fenics
```

Due to network reasons, downloading directly from the PPA can result in lost content. Here, the reverse proxy of USTC is introduced. Backup fenics-packages-ubuntu-fenics-jammy.list

```
sudo cp /etc/apt/sources.list.d/fenics-packages-ubuntu-fenics-jammy.list /etc/apt/sources.list.d/fenics-packages-ubuntu-fenics-jammy.list.bak
```

Then copy it to the user folder

```
cp /etc/apt/sources.list.d/fenics-packages-ubuntu-fenics-jammy.list ~/fenics-packages-ubuntu-fenics-jammy.list
```

Open the file

```
code ~/fenics-packages-ubuntu-fenics-jammy.list
```

Change the file to

```
deb https://launchpad.proxy.ustclug.org/fenics-packages/fenics/ubuntu jammy main
# deb-src https://launchpad.proxy.ustclug.org/fenics-packages/fenics/ubuntu jammy main
```

Then replace the original file

```
sudo cp ~/fenics-packages-ubuntu-fenics-jammy.list /etc/apt/sources.list.d/fenics-packages-ubuntu-fenics-jammy.list
```

Finally update

```
sudo apt update
```

to complete the PPA addition operation.

1.6 Installing FEniCSx and Other Components

In bash, execute

```
sudo apt install build-essential
```

to prepare for the following steps. Then execute

```
sudo apt install fenicsx
```

to start installing FEniCSx.

Moreover, in many FEniCSx examples, `pyvista` is required, which needs to be installed using pip3. However, due to network factors, the installation speed is slow, so it is necessary to switch to a domestic source, such as the Tsinghua source.

In bash, execute

```
python3 -m pip install -i https://pypi.tuna.tsinghua.edu.cn/simple --upgrade pip
```

to upgrade pip3. Then execute

```
python3 -m pip config set global.index-url https://pypi.tuna.tsinghua.edu.cn/simple
```

to switch all sources to the Tsinghua source. Next, execute

```
python3 -m pip install pyvista
```

to install `pyvista`.

In the current FEniCSx tutorial, the use of ADIOS2 is also involved, which is mainly used for post-processing with Paraview. According to the discussion on Github, it needs to be installed before FEniCSx. The document provides a method for installation from the source, which interested users can study on their own.

1.7 Test Installation

Execute in bash

```
python3 -c 'import␣dolfinx'
```

If there are no error prompts, the installation is successful. Next, you can download demo_poisson.py or directly copy the local file with the same name

```
cp /usr/share/dolfinx/demo-python/demo_poisson.py ./
```

And compile in bash

```
python3 demo_poisson.py
```

to check the result.

Chapter 2

Installing FEniCSx in Docker

Most of the content in this section comes from the installation tutorial on Github, with some details being the accumulation of personal experience.

2.1 Install WSL and Docker

For Windows 11 users, you need to install WSL before using Docker, and by default it's WSL 2. The specific steps can be referred to in section 1.1.

After installing WSL, users can download Docker Desktop. Install and restart to start Docker's learning process. Note that in the current version of Windows, Docker cannot change the installation path by default. More detailed content can be seen here.

2.2 Run FEniCSx Container

The FEniCSx team has pre-built the Docker image for FEniCSx. Since the DOLFINx Docker image is hosted on Docker-hub, users can directly access this image in cmd

```
1  docker run -ti -v D:/work-fenicsx:/root/shared --name fenicsx-container dolfinx/dolfinx:stable
```

In the code above, the existing work path `D:\work-fenicsx` on the computer is set as the shared path, and the container name is `fenicsx-container`.

Adding a shared path is due to Docker's own characteristics, it cannot access files outside the image by default. If there are already written files in the work path `D:\work-fenicsx`, it is necessary to share the existing work path into the Docker container.

2.3 Query Container

In cmd, execute

```
1  docker ps
```

to see the information of the currently running containers. Execute

```
1  docker ps -a
```

to query the information of all containers.

2.4 Exit, Stop, Re-enter, and Even Delete Container

Like general command line operations, in the Docker container

```
fenicsx@d66e6f16a673:~
```

Enter

```
exit
```

to exit the container.

If you want to stop the still-running container, you can execute in cmd

```
docker stop d66e6f16a673
```

or

```
docker stop fenicsx-container
```

to stop the existing container. The former is the container ID, the latter is the container name.

If you want to re-enter the container, you need to start the container first. Just execute in cmd

```
docker start fenicsx-container
```

Then execute

```
docker exec -ti fenicsx-container /bin/bash -l
```

to enter the container.

If the container is no longer needed, execute in cmd

```
docker rm fenicsx-container
```

to delete it directly.

2.5 Pulling Images

As FEniCSx is still in development, it is often necessary to pull images to use the latest version. Execute the following in cmd:

```
docker pull dolfinx/dolfinx
```

The system will pull the latest content locally. Then, execute

```
docker image list
```

to see all local image information.

2.6 Compiling Files

Download demo_poisson.py to `D:\work-fenicsx`, enter the container, and execute in the `shared` folder:

```
python3 demo_poisson.py
```

to compile the file.

2.7 Installing Other Components

After running the above example, the system will display

```
pyvista is required to visualise the solution
```

This is because `pyvista` is not successfully installed. According to the answer on fenicsproject.discourse.group, the process of installing pyvista is summarized.

First, switch sources. Copy `sources.list` to the `shared` folder:

```
cp /etc/apt/sources.list ~/shared/sources.list
```

Then change the contents of `sources.list` in `D:\work-fenicsx`, see section 1.4. Copy the modified file back:

```
cp ~/shared/sources.list /etc/apt/sources.list
```

Now start the installation, execute the following in order:

```
export PYVISTA_OFF_SCREEN=true
apt update
apt install -y --no-install-recommends libgl1-mesa-dev xvfb
pip3 install pyvista
```

to complete the installation. As stated in the forum, executing

```
python3 demo_poisson.py
```

again will result in a `u.png` file.

If you need to use ADIOS2, according to the documentation, users can install it through Docker. Interested users can explore it on their own.

Chapter 3

Installing FEniCSx in Anaconda

This part is mainly sourced from Github.

3.1 Installing WSL and Anaconda

Installing FEniCSx in Anaconda cannot be directly achieved on Windows 11, as some dependencies do not support the Windows operating system, so it still needs to be installed in WSL. Specific steps can be referred to in section 1.1.

After installing WSL, download the Anaconda 64-Bit (x86) installation file `Anaconda3-*-Linux-x86_64.sh`, where `*` represents time.

After the download is completed, enter the directory where `Anaconda3-*-Linux-x86_64.sh` is located in bash, copy it to the user folder:

```
cp Anaconda3-*-Linux-x86\_64.sh ~/
```

Enter the user folder and install:

```
cd ~/
bash Anaconda3-*-Linux-x86\_64.sh
```

Next, follow the system prompts to install, where there are several options. The first one is:

```
Do you accept the license terms?
```

This must be answered with `yes`. The second one is:

```
Anaconda3 will now be installed into this location:
/home/USERNAME/anaconda3
- Press ENTER to confirm the location
- Press CTRL-C to abort the installation
- Or specify a different location below
```

Here `USERNAME` is the username, it is recommended for users to install at this path. The third one is:

```
Do you wish the installer to initialize Anaconda3
by running conda init? [yes|no]
```

It's recommended to answer `yes`, otherwise users need to add the environment variable themselves. After the installation is complete, close and reopen bash, if `(base)` is displayed before the path, it means the installation was successful. If users do not wish for `base` to be enabled, execute the following in bash:

```
conda config --set auto_activate_base false
```

3.1.1 Using Mirror Image

Due to the limitations of the network in mainland China, users can also access Anaconda's mirror images in mainland China, such as those from Tsinghua University and Peking University, to download the installation files.

After the installation is completed, open the `.condarc` file in the user directory and modify its content. For example, use VS Code[1]

```
code ~/.condarc
```

Copy the following content to the file[2]

```
channels:
- defaults
show_channel_urls: true
default_channels:
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/r
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/msys2
custom_channels:
conda-forge: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
msys2: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
bioconda: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
menpo: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
pytorch: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
pytorch-lts: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
simpleitk: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
```

Save and exit, then execute the following command to clear the index cache to ensure that the index provided by the mirror site is used.

```
conda clean -i
```

3.1.2 Using a Custom Virtual Environment

When using Anaconda, it defaults to the `base` environment, but users can also customize environments. In fact, the FEniCSx installation manual suggests installing in a virtual environment.

To do this in `base`, execute

```
conda create -n fenicsx-env python=3.10
```

This means creating a virtual environment named fenicsx-env with python version 3.10 (the version can be unspecified). To activate the fenicsx-env virtual environment, execute

```
conda activate fenicsx-env
```

To exit the activated virtual environment, execute

```
conda deactivate
```

To delete the fenicsx-env virtual environment, execute

```
conda remove -n fenicsx-env --all
```

[1] See details in Section 1.3
[2] from Tsinghua University's mirror

3.2 Installing FEniCSx

After installing Anaconda, execute the following in bash:

```
conda create -n fenicsx-env
conda activate fenicsx-env
conda install -c conda-forge fenics-dolfinx mpich pyvista
```

to install.